### תיאור קצר של הפיצ'רים שבחרנו לממש בתרגיל הקודם:

**פיצ'ר 1 : מציאת עבודה**

מאפשר למשתמש למצוא אנשי קשר שעובדים בתעשיית ההייטק.

המערכת בודקת בבסיס נתונים, האם בתיאור העבודה של חבר של המשתמש מופיעות אחת או יותר מילות מפתח המקושרות להייטק, כאשר במידה וכן המערכת מציגה את שמו, תמונתו ומקום עבודתו.

פיצ'ר זה נמצא במסך FIND A JOB (בקוד – במחלקת JobPanel).

**פיצ'ר 2 : מציאת בן זוג**

מאפשר למשתמש למצוא בני זוג אופציונאלים המתאימים למשתמש לפי פרמטרי ההעדפה שלו (טווח גילאים רצוי, העדפה מינית).

המערכת בודקת עבור כל חבר האם הוא עומד בפרמטרים הללו, בנוסף בודקת האם מצב מערכת היחסים שלו מתאים ובמידה וכן מציגה למשתמש את תמונת ושם החבר. לאחר לחיצה על אחד המועמדים, המערכת מציגה פרטים בסיסיים על אותו חבר כמו גם את אלבומיו, תמונותיו השונות ואף את מרחקו בקילומטרים מן המשתמש.

פיצ'ר זה נמצא במסך FIND A MATCH (בקוד – במחלקת MatchPanel).

### תבנית מס' 1 – FACADE

* סיבת הבחירה / שימוש בתבנית:

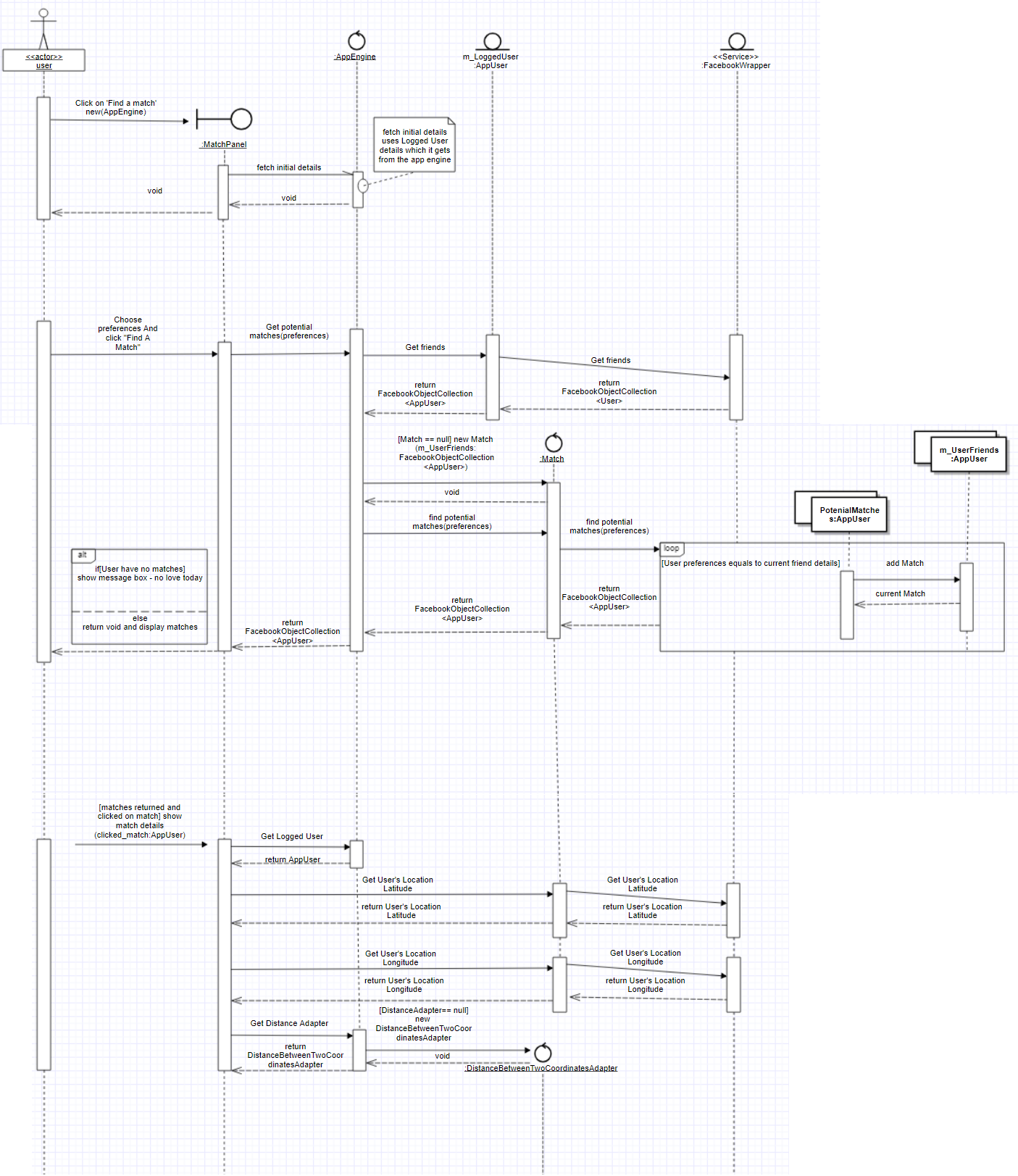
איגוד כל הלוגיקה ותתי המערכות של האפליקציה והנגשתה לממשק המשתמש בצורה פשוטה.

המחלקה הנ"ל מהווה חוצץ בין הלוגיקה לממשק המשתמש, ולכן ראינו לנכון לממש אותה כFACADE.

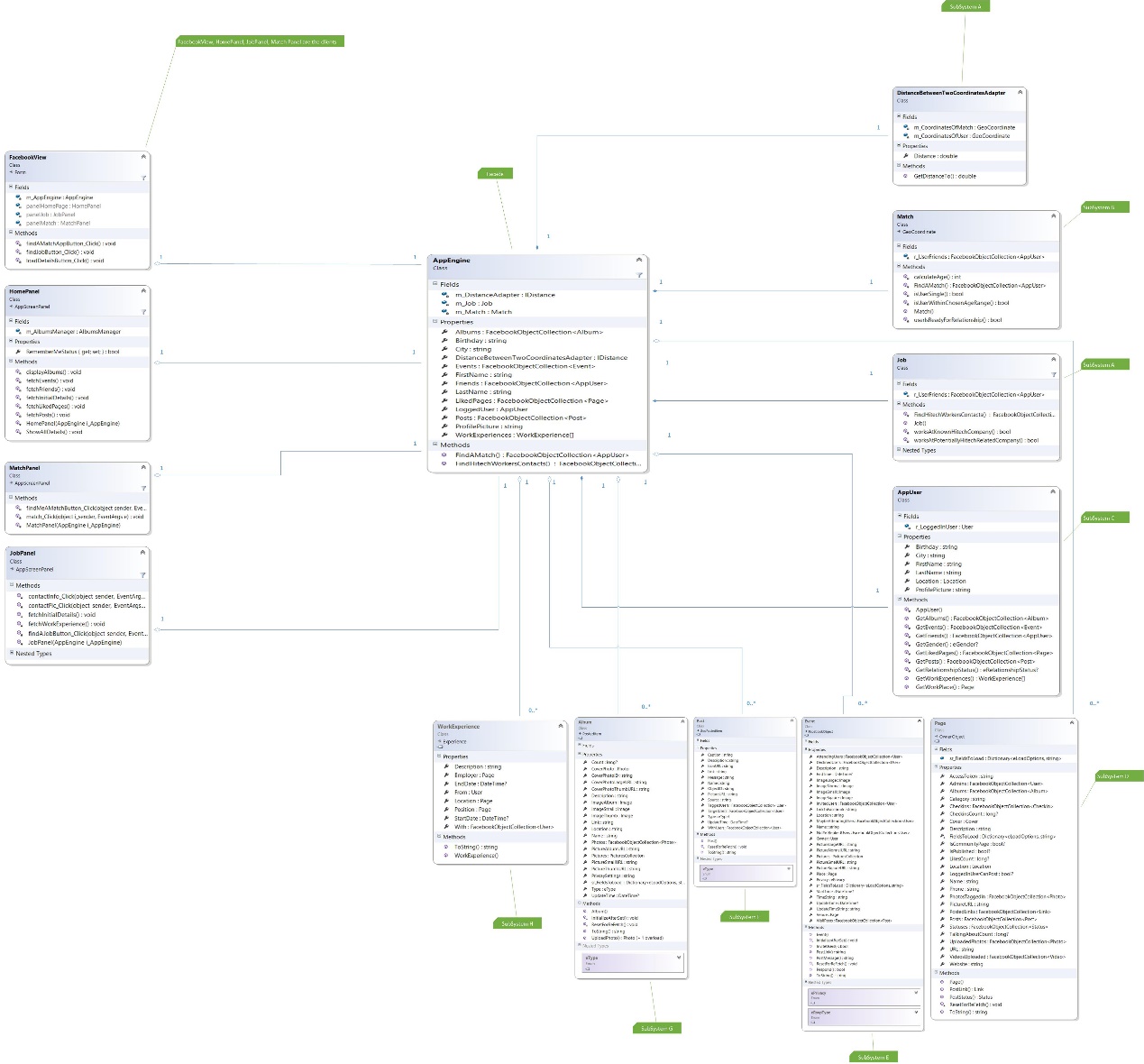
* אופן המימוש:

מדובר במחלקה AppEngine כאשר השירותים שהיא חושפת למשתמש הם שימוש פשוט בשני הפיצ'רים על ידי הפעלת מתודות FindHitechWorkersContacts, FindAMatch ושימוש במידע על המשתמש הרלוונטי לאפליקציה באמצעות properties מתאימים.

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – FACTORY METHOD

* סיבת הבחירה / שימוש בתבנית:

האפליקציה מרובת מסכים – כלומר בטופס אחד בכל פעם מורכב סוג של AppScreenPanel אחר המייצג מסך באפליקציה, כאשר ל AppScreenPanels השונים יש מאפיינים משותפים ועל כן מהווים משפחה פולימורפית ולכן יצרנו Factory Class סטטי אשר מבצע את מלאכת היצירה ולוגיקת בחירת האובייקט מהמשפחה הפולימורפית במתודה סטטית שלו כיוון שאנו שואפים להימנע מלממש זאת בClient.

* אופן המימוש:

המשפחה הפולימורפית ממומשת בקוד בצורה הבאה:

AppScreenPanel – מחלקה אבסטרקטית היורשת מ-UserControl ומהווה בסיס משותף למשפחה הפולימורפית.

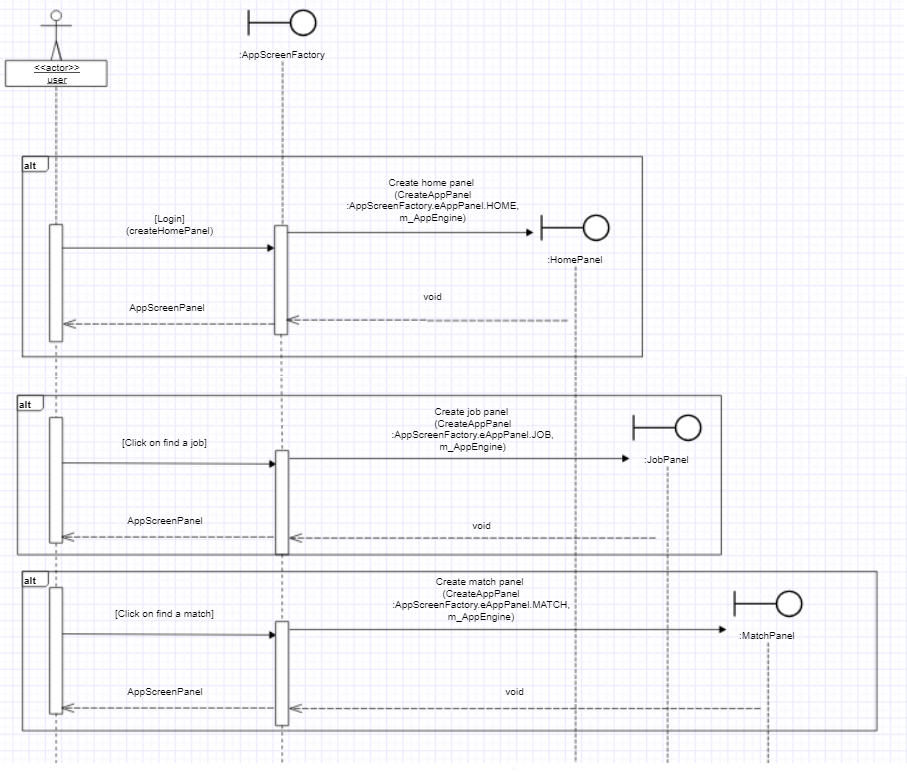
HomePanel – מחלקה המהווה כמסך הראשי של האפליקציה ויורשת מ- AppScreenPanel.

JobPanel – מחלקה המהווה כמסך של הפיצ'ר find a job באפליקציה ויורשת מ-AppScreenPanel.

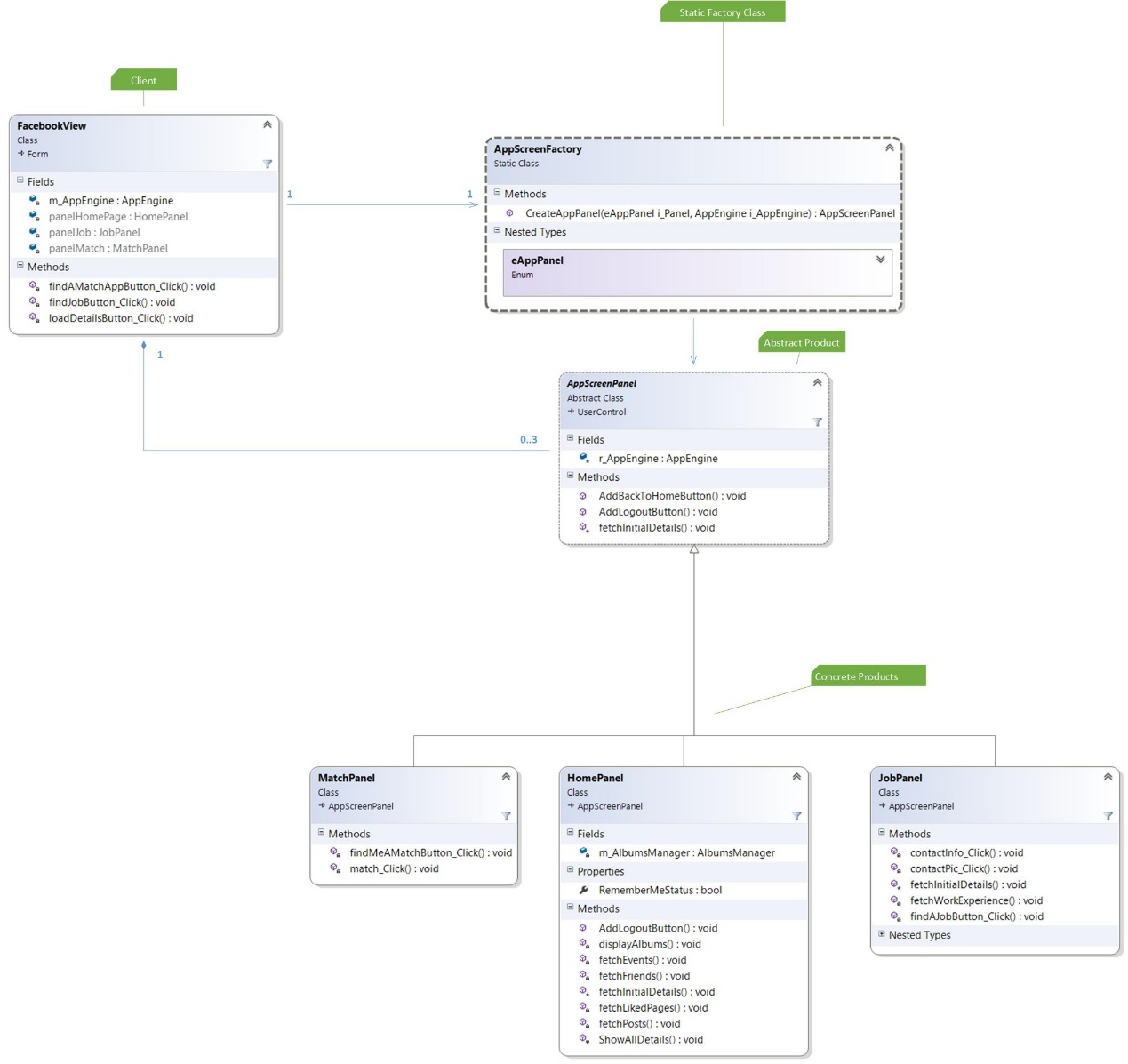
MatchPanel – מחלקה המהווה כמסך של הפיצ'ר find a match באפליקציה ויורשת מ- AppScreenPanel.

FactoryScreenPanel – מחלקה סטטית היוצרת את המסך המתאים בהתאם לContext המסך הרלוונטי.

* Sequence Diagram



* Class Diagram



### תבנית מס' 3 – ADAPTER

* סיבת הבחירה / שימוש בתבנית:

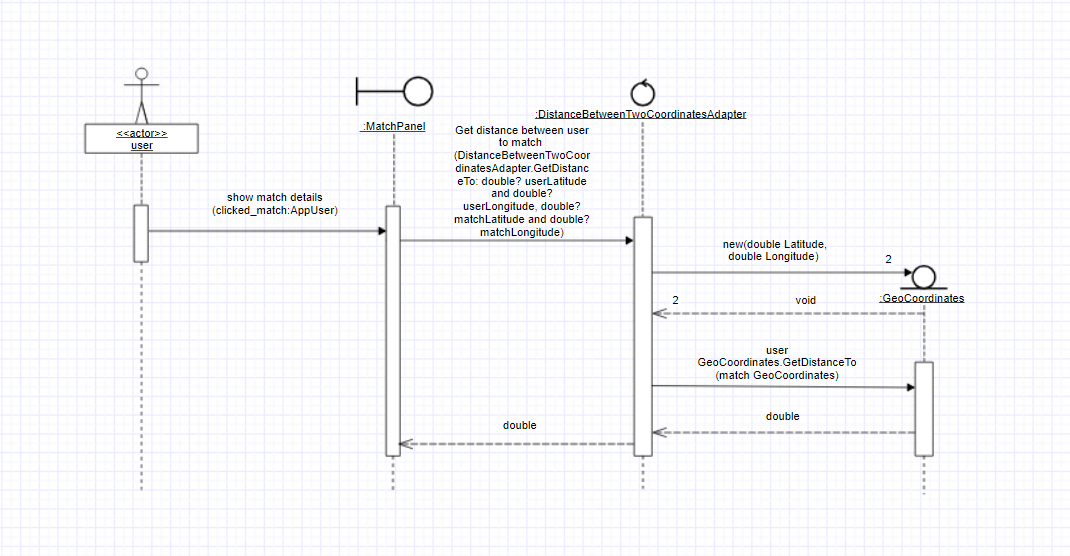
כחלק מהפיצ'ר find a match כאשר לוחצים על התאמה רלוונטית למשתמש מוצג המרחק בקילומטרים בין המשתמש לאותה התאמה.

מרחק זה מחושב על ידי מתודה בשם GetDistanceTo המופעלת על שני אובייקטים מסוג ,GeoCoordinate מתודה זו מהווה LegacyCode כאשר המידע מסופק על ידי Facebook (גם הוא מהווה LegacyCode) אינו בפורמט המתאים, ולכן מימשנו Adapter שיממש את ההתאמה כאשר הClient הוא ממשק המשתמש של האפליקציה והשירות הוא הDistanceTo.

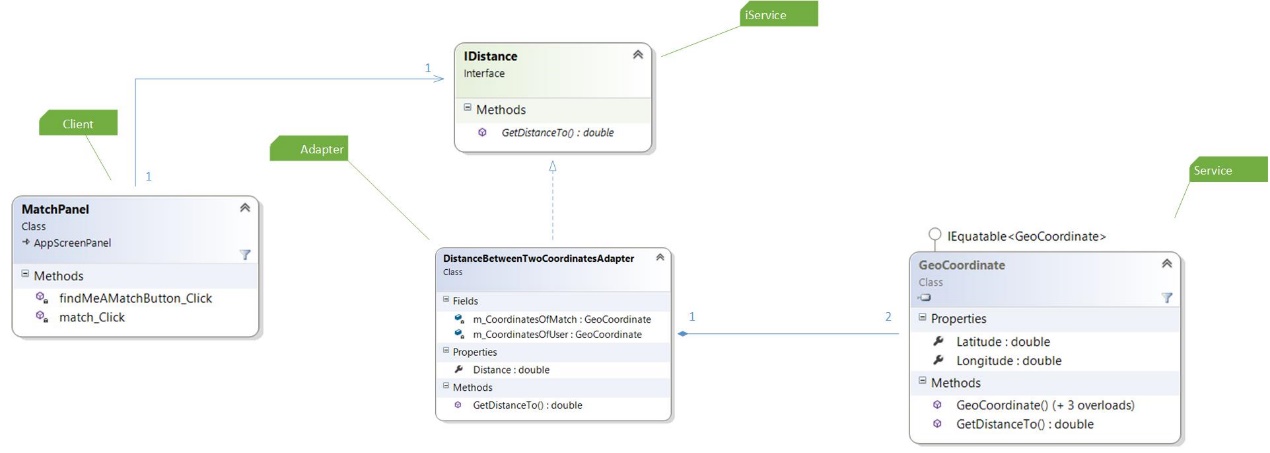
* אופן המימוש:

המחלקה DistanceBetweenTwoCoordinatesAdapter מקבלת כקלט שתי זוגות קורדינטות מסוג double? ומייצרת (במידה ואף אחת מהקורדינטות אינה null) שני אובייקטים מסוג GeoCoordinate ומפעילה את המתודה GetDistanceTo על שני האובייקטים הללו ומחזירה את תוצאתה.

* Sequence Diagram



* Class Diagram



### עבודה עם תכנות א-סינכרוני

* המקומות בהם השתמשנו בתכנות א-סינכרוני הם:
  + FacbookView.loadSettings מדובר במתודה שמופעלת במידה והמשתמש בחר לשמור את העדפותיו ולכן אנו טוענים את כל דף הבית (ע"י מתודה HomePanel.ShowAllDetails) פעולה זו דורשת הבאת מספר רב של נתונים מFacebook ולכן הקריאה ל - HomePanel.ShowAllDetails מופעלת בThread נפרד.
  + AlbumManager.AlbumClick מתודה זו מביאה את כל התמונות מFacebook עבור האלבום הנבחר. מאחר ומדובר בתמונות, פעולת ההבאה וההצגה מתוך המתודה נעשית בThread נפרד.
  + AppScreenPanel.fetchInitialDetails מדובר במתודה המציגה פרטים התחלתיים של המסך בו המשתמש נמצא המובאים מFacebook ולכן מבוצעת בכל מסך בThread נפרד בקונסטרקטור.
  + HomePanel.ShowAllDetails מאחר ומתודה זו מאכלסת נתונים (הנשלפים מ Facebook) במספר פקדים בדף הבית, הבאת הנתונים אל כל פקד (ע"י שימוש במתודות HomePanel.fetchEvent, HomePanel.fetchAlbums, HomePanel.fetchPosts, HomePanel.fetchFriends) נעשתה כל אחת בThread נפרד על מנת שלא תיפגע חווית המשתמש.

בנוסף אנו מאפשרים את טעינת כל הנתונים הנ"ל ע"י לחיצה על הכפתורים המתאימים (Albums, Friends, Posts, Events) כל לחיצה מפעילה את המתודה המתאימה מהנ"ל בThread נפרד.

* + JobPanel.FindAJobButton\_Click מתודה זו מוצאת עבור המשתמש רשימה של אנשי קשר ומציגה את שמם, היכן הם עובדים ואת תמונותיהם (על ידי ,JobPanel.addAllContantsToListBox FriendsDisplayer.Display בהתאמה) כאשר פעולות אלו עלולות לגזול זמן ולפגוע בחווית המשתמש ולכן הקריאות ל - ,JobPanel.addAllContantsToListBox FriendsDisplayer.Display מופעלות כל אחת בThread נפרד.
  + MatchPanel.FindMeAMatchButton\_Click מתודה זו מוצאת עבור המשתמש ומציגה את תמונות הפרופיל של ההתאמות שנמצאו עבורו (על ידי FriendsDisplayer.Display) ומאחר ומדובר בהרבה נתונים הקריאה ל - FriendsDisplayer.Display מופעלת בThread נפרד.
  + MatchPanel.match\_Click מתודה זו מופעלת כאשר לחצו על תמונת התאמה שנמצאה, ושולפת את הנתונים מFacebook עבור אותה התאמה (פרטים אישיים, אלבומי תמונות) ולכן הפעולות הנ"ל נעשות כל אחת בThread נפרד.

### עבודה עם Data Binding

* המקומות בהם השתמשנו בData Binding הם:
  + HomePanel –
    - ListBoxLikedPages מקושר על ידי pageBindingSource לLikedPages של הAppUser המחובר.
    - ListBoxEvents מקושר על ידי eventBindingSource לevents של הAppUser המחובר.
  + JobPanel –
    - ListBoxWorkExperiences מקושר על ידי workExperienceBindingSource לworkExperience של הAppUser המחובר.
  + UserDetailsPanel – מקושר על ידי appUserBindingSource ל AppUser הנבחר

(מופיעה גם בHomePanel וגם בMatchPanel).